RAISONANCE

EvoPrimer for STM8L Ultra-low-power MCU

Fun, easy introduction kit for STM8L microcontrollers

User Manual

Document version
13 July 2010

Contents

1. Introduction....4

1.1 Purpose of this manual....4
1.2 Scope of this manual....4
1.3 Additional help or information....4

2. Presentation....5

2.1 Base features....5
2.2 Target board features....5
2.3 Development software....6

3. EvoPrimer hardware....7

3.1 Package contents....7
3.2 Components overview....7
3.3 Target board features....8
3.3.1 STM8L152C4T6 microcontroller features....8
3.3.2 Target board power supply....8
3.3.3 Target board MEMS accelerometer....8
3.3.4 Target board 128KB I2C EEPROM....9
3.3.5 14-segment LCD....9
3.3.6 Temperature sensor....9
3.3.7 Push button....9
3.4 Base features....9
3.4.1 3D MEMS accelerometer....9
3.4.2 Power supply....9
3.4.3 Extension connector....10
3.4.4 Application-specific extension boards....11

4. Getting started....12

4.1 Switch on....12
4.1.1 Insert the target board....12
4.1.2 Connect battery, charge and power up....13
4.2 Using your EvoPrimer applications....13
4.2.1 Play....13

4.2.2 Standalone operation....14
4.2.3 TempLP application....14
4.2.4 Breakout application....14
4.2.5 Bubble level application....15
4.3 Configuration menu....15
4.4 Compiling, programming and debugging....15
4.5 Install the Ride7 software toolset for STM8....15
4.6 Explore the STM8 Toggle application....16

5. Managing your CircleOS applications....17

5.1 CircleOS architecture....17
5.2 CircleOS Scheduler....17
5.2.1 Initialization stage....17
5.2.2 Periodic systick interrupt....17
5.2.3 Application scheduler....17
5.3 CircleOS resource usage....18
5.3.1 ROM usage....18
5.3.2 RAM usage....18
5.3.3 EEPROM usage....18
5.4 Managing applications on your EvoPrimer....19
5.5 Selecting the current application....20
5.6 Downloading new applications....20
5.7 Restoring the factory configuration....20
5.8 Resetting your EvoPrimer....20

6. Developing CircleOS applications....21

6.1 Developing your first CircleOS application....21
6.2 Libraries....22
6.3 Sharing your application with the Circle community....22

7. Conformity and recycling....23

8. Glossary....24

9. Index....25

10. History....26

1. Introduction

The **EvoPrimer** for STM8L Ultra-low-power line is a fun, cost-effective evaluation and development package based on Raisonance's versatile, innovative **Open4** platform and STM8L152 target board.

Note: The ST product **EvoPrimer** for STM8L Ultra-low-power line (ST order code: STM8L1526PRIMER) is a derivative product of the **Raisonance Open4**. The same Primer target boards, extension boards, software tools and sample applications are used with both the EvoPrimer and Open4.

1.1 Purpose of this manual

This manual provides an overview and installation procedure for your EvoPrimer. Armed with the knowledge in this manual users can quickly understand the target microcontroller's features and create their own applications.

1.2 Scope of this manual

This manual is applicable to all EvoPrimer versions and describes the EvoPrimer's basic use and its hardware and firmware features.

Some technical information about your EvoPrimer is available from the Ride7 interface:

- The 3D MEMS inertial sensor data sheet
- The STM8L 8-bit ultra-low-power MCU documentation
- STM8 MCU reference manual

Visit the Circle community web site at *www.stm8circle.com* for resources and software tool downloads.

1.3 Additional help or information

Please visit the Raisonance website: *http://www.raisonance.com/* and its support forum or contact Raisonance.

Address: Raisonance S.A.S.
17, Avenue Jean Kuntzmann,
38330 Montbonnot Saint Martin
France
Telephone: +33 4 76 61 02 30
Fax: +33 4 76 41 81 68
Email: ***support@raisonance.com***

If you find any errors or omissions, or if you have suggestions for improving this manual, please let us know by email.

2. Presentation

Your EvoPrimer is composed of two major hardware elements:

- a base platform,
- a target board featuring an STM8L152.

The target board can be used both when connected to the base and in Standalone with the demonstration or a user application.

When connected to the base, the STM8L152 drives the hardware features on the base. This target microcontroller can be reprogrammed and application software can be created and debugged using the Ride7 Raisonance software tools when the base is connected to a PC.

The STM8L152 target board's rechargeable battery and solar cells allow this board to be used in a standalone mode with demonstration and users applications such as the provided temperature sensor application.

The following sections describe the features of the base and target board, software tools and fundamental procedures for their use.

EvoPrimer

2.1 Base features

The base provides the following capabilities:

- USB debug/programming connection (SWIM for STM8)
- QVGA (320x240) backlit color LCD, with touch screen capability
- Joystick/push button (mechanical)
- 4 push buttons (based on touchscreen capability)
- Micro SD card connector
- Audio circuit with Microphone/Loudspeaker/Jack (buzzer capability only)
- 3D MEMS accelerometer
- Li-Ion battery with charge management circuitry
- Add-on connector (USART, SPI, I2C, ADC,...)

2.2 Target board features

The STM8L152 target board integrates the following capabilities:

- STM8L152 microcontroller (32KB Flash, 2KB RAM, 1KB EEPROM)
- LCD (14-segment)
- Temperature sensor
- MEMS accelerometer
- 128KB I2C EEPROM (can hold 4 CircleOS applications)
- Li-Ion battery
- Solar cells
- Supports standalone operation and can take advantage of its solar cells to perform long-haul tasks in low-power mode.
- When connected to its base, it can take full advantage of the available features.

2.3 Development software

The development software offered for the EvoPrimer for STM8L provides everything you need to program the STM8L and debug applications, including:

- USB host connection for in-circuit programming and debugging.
- Ride7 integrated development environment for code editing, device programming and application debugging.
- Raisonance optimizing C compiler toolchain for STM8.
- CircleOS Operating System and base services. An online community available at *www.stm8circle.com* makes it possible to share your EvoPrimer experience with others on the forums, retrieve useful resources, application notes, firmware, demos (C sources and projects) and applications, all available for free download.

3. EvoPrimer hardware

3.1 Package contents

Your EvoPrimer unit has been carefully packed. Examine the equipment for damage that may have occurred during shipment. If you find any damage, or if any of the items are not included, please contact Raisonance.

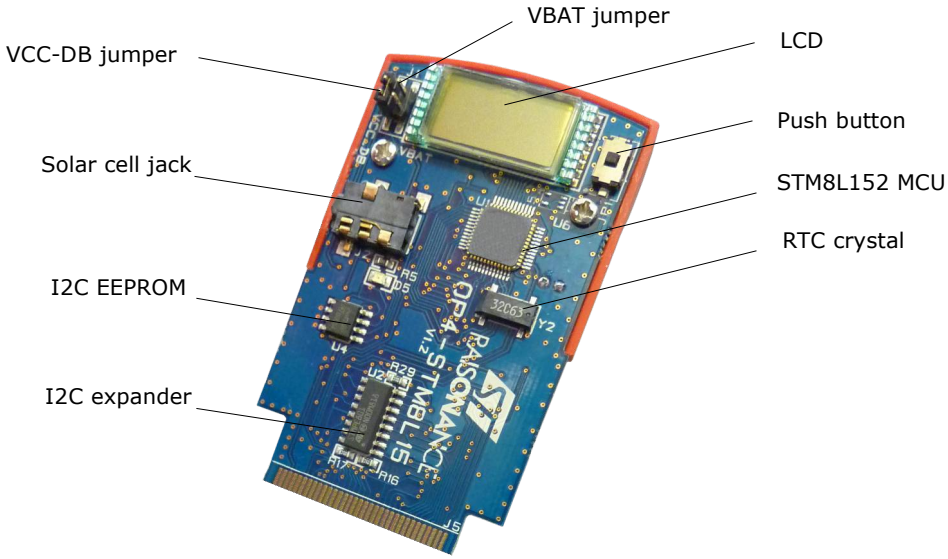
Your EvoPrimer is delivered in two packages:

1. The EvoPrimer base with USB cable,
2. The STM8L152 target board with solar cells on a support pedestal.

Note: Download software tools after free registration at *www.stm8circle.com*

3.2 Components overview

Here is a brief overview of the main components of the STM8L152 Target Board:



3.3 Target board features

3.3.1 STM8L152C4T6 microcontroller features

The target board is equipped with an STM8L152C4T6, from STMicroelectronics' STM8L ultra-low-power line of 8-bit microcontrollers. Its main characteristics are:

- Advanced STM8 core, Harvard architecture with 3-stage pipeline, 16 CISC MIPS peak
- 32KB of Flash program memory, 2KB RAM, 1KB EEPROM with ECC and RWW
- Embedded oscillators (16 MHz)
- SWIM debug interface
- Low-power RTC
- Integrated LCD controller (4x28 segments)
- 4 DMA channels
- 12-bit DAC with output filter
- 12-bit ADC up to 1Msps, 25 channels
- 4 timers, 2 watchdogs
- Up to 41 I/Os, all mappable to interrupts
- Embedded communication peripherals: USART, SPI, I2C, IrDA
- 96-bit unique ID

3.3.2 Target board power supply

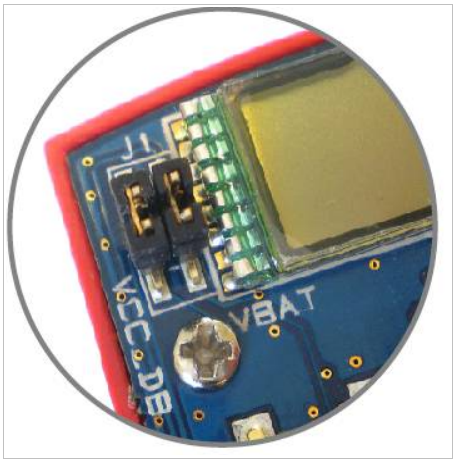
The STM8L152 target board includes its own Li-Ion battery to support use in standalone mode.

This battery can be recharged by the solar cells on the support pedestal, when this support is connected to the jack on the target board. This makes it possible to use the target board to perform long haul tasks in low-power mode.

The target board battery may also be recharged by the battery or USB connection of the base.

Power supply jumpers on the target board connect/disconnect the target board's battery (VBAT jumper) and power on/off the STM8L152 (VCC_DB).

Remove these jumpers when the target board is not in use to avoid fully discharging the target board's battery.



Note: It is recommended to charge the battery for 30 hours before using it.

This operation must be performed with the target board inserted into the base and connected to the USB port to a PC (ensure the PC does not go into sleep mode during this period).

Warning: If the target battery is ever fully discharged, it is **impossible to recharge** it. To avoid this, remove VCC_DB jumper whenever the target board is not used and not connected to the base or solar cells.

3.3.3 Target board MEMS accelerometer

The target board is equipped with a MEMS inertial sensor (LIS3LV02DL from STMicroelectronics). which can be used by demonstration or user applications when on the base or in standalone mode.

3.3.4 Target board 128KB I2C EEPROM

To complement the on-chip Flash of the STM8L152, the target board also includes a 128 Kbyte I2C EEPROM (STMicroelectronics' M24M01-HR). When using the EvoPrimer with CircleOS, this additional memory makes it possible for the EvoPrimer to manage up to 4 CircleOS applications.

3.3.5 14-segment LCD

The STM8L152 target board includes its own 14-segment LCD display which is driven by the STM8L152 and may be used in demonstration and user applications. In the temperature sensor demonstration application, it is used for temperature display when mounted on the base or when used in standalone mode.

3.3.6 Temperature sensor

The STM8L152 target board includes its own temperature sensor (STMicroelectronics' STLM20W87F) which can be used by applications when on the base or in standalone mode.

3.3.7 Push button

The STM8L152 target board includes its own push button which can be used by applications when on the base or in standalone mode.

3.4 Base features

3.4.1 3D MEMS accelerometer

The base is equipped with a MEMS inertial sensor (LIS3LV02DL from STMicroelectronics) which can be used to select commands in coordination with a graphic pointer.

When you start the EvoPrimer for the first time, you will see a small ball moving according to the orientation of the EvoPrimer. This 3D positioning information is provided by the MEMS.

3.4.2 Power supply

The base features a 400mAh Li-Ion rechargeable battery, equipped with a voltage regulator and a battery charger.

When the USB connector is linked to a PC, the host voltage (supplied by the PC) recharges the battery.

When no USB host is connected, the battery supplies the power for the EvoPrimer.

When the battery is fully charged, the EvoPrimer can be used for about 6 hours. The duration of the batteries depends on the activity.

3.4.3 Extension connector

The base has a 20-pin HE14 female right-angle extension connector that you can access when removing the orange part of the case. The extension board mechanical description is available in the *www.stm8circle.com* resources section. The extension connector permits usage of the following pins:

Pin	Printed name	STM8L pin	STM8L name	Description
1	V2V8	Vcc	VCC	Connected to U6 regulator output (3.1V). Can provide approximately 100mA to the extension board.
2	GND	GND	GND	Ground
3	SCL	PC.1	I2CSCL	I2C clock
4	SDA	PC.0	I2CSDA	I2C data
5	MISO	PB.7	SPI_MISO	SPI MISO
6	SD	PB.6	SPI_MOSI	SPI MOSI
7	SCK	PB.5	SPI_SCK	SPI Clock
8	WS	PE.3	PE3	GPIO Port E bit 3
9	CANH	PE.4	PE4	GPIO Port E bit 4
10	CANL	PE.5	PE5	GPIO Port E bit 4
11	ADC1	PD.0	PD0	GPIO Port D bit 0
12	ADC2	PD.2	PD2	GPIO Port D bit 2
13	A_TIM	PD.3	PD3	GPIO Port D bit 3
14	CTS	PB.0	PB0	GPIO Port B bit 0
15	RTS	PB.1	PB1	GPIO Port B bit 1
16	TX	PA.2	TX	UART Tx
17	CK		PBUTTON	Push button input. Can wakeup the target board.
18	RX	PA.3	RX	UART Rx
19	VEXT		VBAT	Battery voltage input for the extension board
20	GND	VSS	GND	Ground

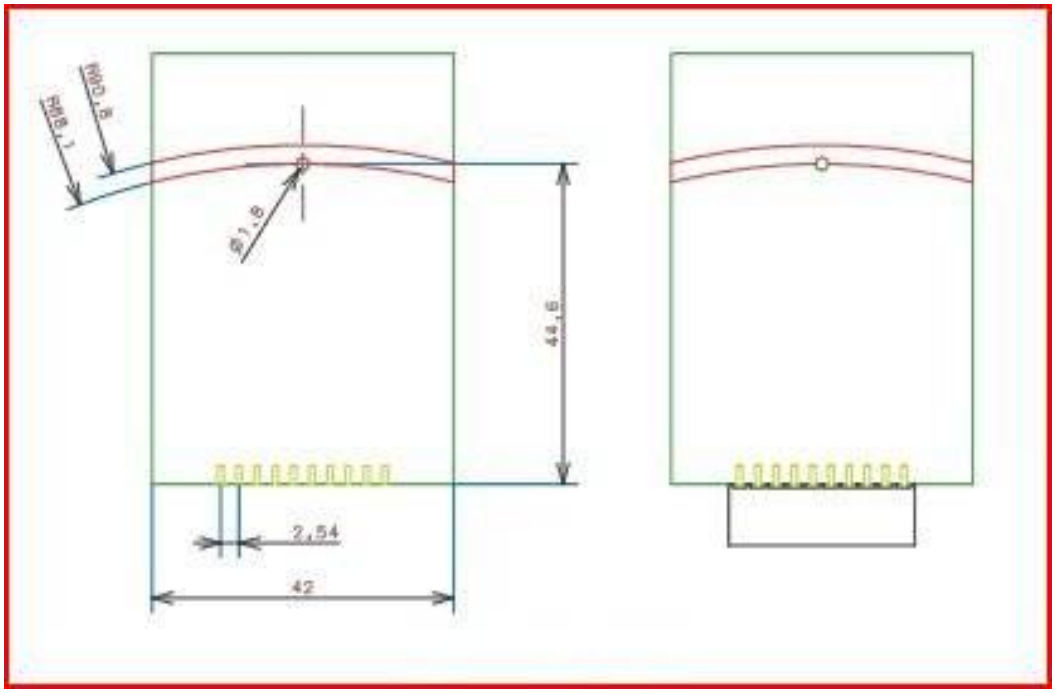
Note: Printed names on the extension boards may not accurately match the STM8L target board names as the generic base supports other processor families (namely STM32) with other features.

3.4.4 Application-specific extension boards

The base has been designed to allow extension through application-specific extension boards.

Such extension boards connect to the extension connector and are compatible (mechanically and electronically) with the previous STM32-Primer2, so you should be able to reuse your previous designs without modifying the extension board. However, some limitations may apply, for instance if you use specific processor/peripheral capabilities that are not portable between designs.

You can design your own extension board if your application requires it. The following picture shows the mechanical constraints that you should respect so that your extension board fits inside casing of the base platform.



Extension board mechanical specifications

4. Getting started

This chapter explains how to switch on your EvoPrimer and describes some of the provided applications.

4.1 Switch on

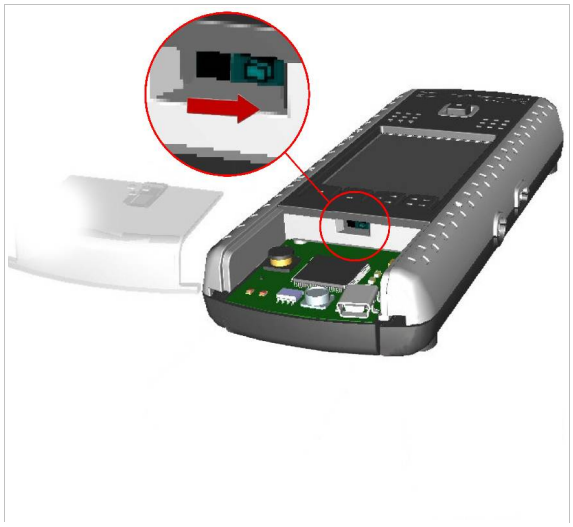
The procedure for getting started is reduced to four simple operations:

1. Insert a target board into the EvoPrimer base (see below).
2. Switch the power switch to provide current to the target board (see below).
3. Slide the transparent cover in place.
4. Press the joystick to power up the EvoPrimer.

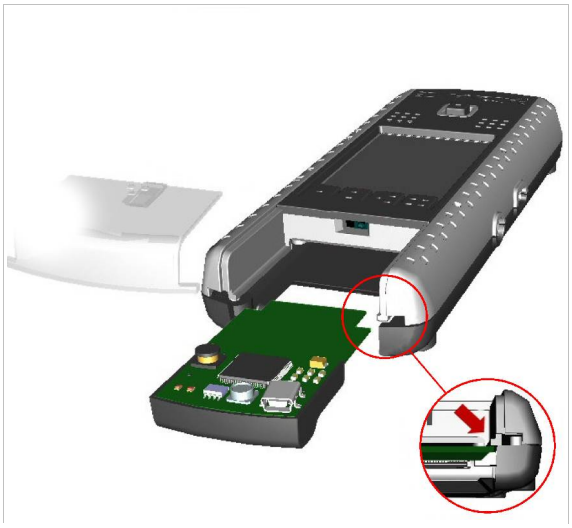
4.1.1 Insert the target board

Follow these steps to insert the target board:

1. Remove the transparent cover.
2. Switch off the battery (see image below left).
3. Insert the target board into the base, as shown in the image, ensuring that the PCB is against the guides.
4. Push until the top of the target board fits with the side of the base.



Switching OFF the Primer

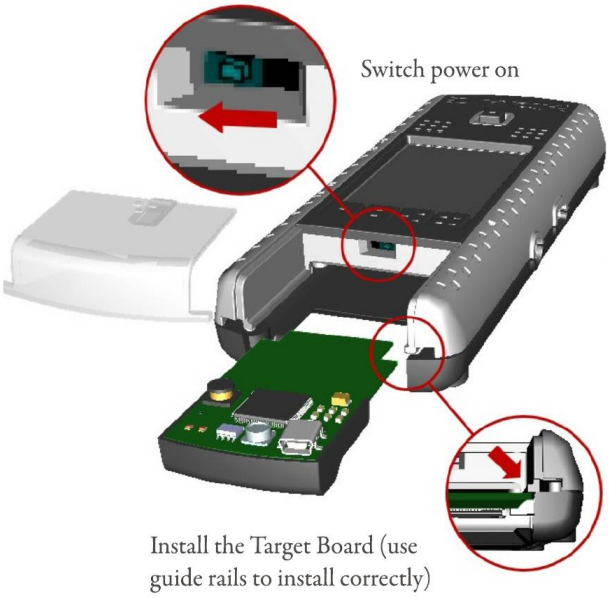


Inserting the target board

4.1.2 Connect battery, charge and power up

Your EvoPrimer is shipped with the battery disconnected, to prevent discharging and potential harm to the battery. You must connect the battery before operating your EvoPrimer:

1. Open the EvoPrimer's case by removing the cover which is on the target board.
2. Switch the button as shown on the image to power-on your device.
3. Close the case.
4. Connect the EvoPrimer to the USB port on a PC to charge its battery. Only the red LED (L0) is lit if the battery is charging.
   The green LED (L1) is lit when fully charged.
5. After charging the battery, press the push button to activate the EvoPrimer.



4.2 Using your EvoPrimer applications

The base provides user interface features such as a color LCD, audio circuit, joystick/push button, GUI, SD card connector, MEMs accelerometer and more. The target board operates evaluation features when connected to the base, and can be a standalone temperature sensor application.

4.2.1 Play

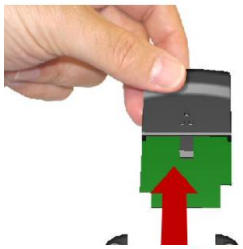
After the opening screen is displayed, press the push button to call the main menu.

Tilt the EvoPrimer backward/forward or use the joystick to navigate in the menu, use the push button to select a menu command. Initial menu items include:

Config	Configure parameters for the EvoPrimer including the pointer, backlight for the LCD display and time
Breakout	This is an addictive game provided as an example. Full source is available from www.stm8circle.com.
Applic	Select an application to run. The application that you choose will then appear in the main menu.
About	Display information about the CircleOS version and available memory.
Shutdown	Turn off the EvoPrimer. To restart the EvoPrimer, simply press the push button.
Quit	Quit the menu

Note: When you receive the EvoPrimer, the MEMS based controls are calibrated to a "zero" position that matches a 30° angle from the horizontal (corresponds to the position of a book when reading). To practice controlling the EvoPrimer, move the blue dot around the main screen. It takes a little practice.

4.2.2 Standalone operation



Some applications are designed to run on the target board in standalone mode. In this style of operation the target board is powered by its own battery with recharging provided by the solar cells. The TempLP application is one such example, once it has been loaded from the I2C EEPROM, it can be run on the target board in standalone mode.

These steps describe the procedure to use it in standalone mode:

1. Power-off the base.
2. Remove the transparent cover that is over the target board.
3. Turn the base so the display is facing downwards.
4. Hold the base with both hands and push upward on the target board with both thumbs pressing evenly on the target boards plastic drawer.
5. When the target board dislodges from the connector, remove it completely by pulling on the plastic drawer.
6. Remove then put back the VCC_DB jumper to start the application.
7. Connect the solar cells as shown here to recharge the target board's battery. Make sure that a light source is available to recharge the target board battery.

Follow steps 1-5 above whenever you remove any target board from the base.

Note: The LCD screen contrast varies depending on the power supply, so in standalone mode the screen contrast may be low if the battery charge is low.

The solar cells are designed to work with natural light. It is ideal to place the target board near a window during daylight for charging.

Important: If the target board battery is ever fully discharged **it is impossible to recharge it**. To avoid this, **remove the VCC_DB jumper** whenever the target board is not connected to the base or the solar cells.

4.2.3 TempLP application

The TempLP application uses the target board's temperature sensor to display the current temperature, and to record the temperature on a regular basis (selectable from 1 second up to 1 hour). The application runs in standalone mode (with the target board disconnected from the base), taking advantage of the very low power mode of the STM8L152. Use it as follows:

1. Insert the target board into the base. Ensure the PCB is against the guides.
2. Set VCC_DB and VBAT jumpers.
3. Switch on the power switch and slide the transparent cover into place.
4. Press the joystick to power up the base. If the EvoPrimer does not start, remove and then put back the VCC_DB jumper to restart the STM8L.
5. Press the joystick to access the main menu.
6. Select **TempLP** from the **Applic** menu to load the application.
7. After loading, select **TempLP** in the main menu to launch the application. The mini LCD displays the current temperature.
8. To configure measurement interval, press the joystick and click on **Select Period**. The STM8L monitors the temperature at the selected interval and refreshes the mini LCD.

4.2.4 Breakout application

Start the Breakout application by selecting it from your EvoPrimer main menu (just press the button). Simply tilt and pan your EvoPrimer to move the paddle appropriately to break all the bricks with a ball.

4.2.5 Bubble level application

From the main menu, select the **Applic**. Entry, then select one of the applications which are stored on your EvoPrimer application storage EEPROM. Select the **BubLevel** application and start it. This application displays a bubble level, using the 3D MEMS to compute the device's horizontal level.

4.3 Configuration menu

Configuration and test applications are pre-installed in order to check your EvoPrimer's capabilities.

From the main menu of the EvoPrimer, run the Config command to access the following parameters:

1. **Interface**:
   User input: Selects the user input for commands menu (accelerometer or joystick or accelerometer + joystick or touchscreen)
   Loudspeaker: Ena/Disables loudspeaker. If not, sounds are sent to jack headphone connector.
2. **Autorun**: Lets the current application run whenever your EvoPrimer is started. In this mode, there is no need to select the application in order to start it.
3. **Time**: Sets your EvoPrimer time. Note: RTC clock remains valid even if EvoPrimer is shut-off.
4. **Test**: Performs a quick factory test of the EvoPrimer.

You can add more applications to your EvoPrimer by downloading them on the web site *www.stm8circle.com/projects*. Refer to Chapter 5 Managing your CircleOS applications.

4.4 Compiling, programming and debugging

To explore, modify and create new applications, use the Raisonance C compiler and Ride7:

1. Register on the *www.stm8circle.com* web site (this is free).
2. Download the CD-ROM-Image file on the "Resources" page.
3. Unzip the files and run the installation, as described in section 4.5.
4. Launch Ride7.
5. Connect your PC to the Debug USB port on the base.

Complete tool documentation is provided with the software installation.

Note : 32 KB output. Licence available for free at "*www.mcu-raisonance.com/stm8_registration.html*"

4.5 Install the Ride7 software toolset for STM8

You must install Ride7 and Rkit-STM8 before you can program or debug applications with your EvoPrimer.

1. Ride7, the Raisonance IDE includes the RLink driver which pilots the programming and debugging of your EvoPrimer from your PC.
2. RKit-STM8, the Ride7 add-on that contains the Raisonance optimizing C compiler toolchain for STM8. A free licence allowing output of code up to 32KB of code is available at "*www.mcu-raisonance.com/stm8_registration.html*" (requires registration). RKit-STM8 contains some utilities to manage your CircleOS applications (CircleOS is the OS embedded on your EvoPrimer; refer to Chapter5.1 "CircleOS architecture" for details).

RKit-STM8 includes many datasheets and manuals, accessible from Ride7 "Documentation" window.

Note: Ride7 and RKit-STM8 should be installed **before** connecting to the debug USB port.

4.6 Explore the STM8 Toggle application

A Toggle example is installed with the RKit-STM8 for Ride7 in the following directory:
"[RIDE7_INSTALL_DIR]\Examples\STM8\STM8L-EvoPrimer\Toggle_with_CircleOS\"
Follow these steps to execute the example on your EvoPrimer:

1. Open Ride7.
2. Select **Open | Project** and navigate to the example.
3. Click on **Project | Make Project**.
4. Once the project is built, connect your EvoPrimer to your PC through USB. Click **Debug | Start**.
5. Once the debugger is ready, you can run the example through **Debug | Run**.

You can then play with the sample application on your EvoPrimer.

5. Managing your CircleOS applications

5.1 CircleOS architecture

Your EvoPrimer is equipped with an STM8L152C6 that contains 32KB of Flash ROM and 2KB of RAM. CircleOS is compiled as a monolithic application on the EvoPrimer, containing the CircleOS code as well as your application code. As the resources are very limited, it is not possible to have more than one application in Flash at a given time.

To workaround this limitation, a 128KB I2C EEPROM is embedded on the STM8L152 target board. Up to 4 images of the Flash can be stored on the EEPROM, and CircleOS has the capability to overwrite the contents of its Flash with one of the images retrieved from the EEPROM. The EvoPrimer embeds the CircleOS operating system (source files are available from *http://www.stm8circle.com/* after free registration). It provides services that help you develop your EvoPrimer applications, including:

- Application management, Menu functions, Scheduler tasks, RTC
- LCD graphic functions, LED functions
- MEMS, Buzzer and Push Button functions.

Each application is run by CircleOS and has the full availability of the CPU, and all the RAM that is not being used by CircleOS. It is scheduled by the CircleOS with full privileges on the device, until it explicitly quits.

You will find details about programming CircleOS applications in Chapter 6.

5.2 CircleOS Scheduler

CircleOS acts in several stages: An initialization stage which occurs upon device reset, a periodic systick interrupt, and the scheduling of applications.

5.2.1 Initialization stage

During the initialization stage, the hardware configuration is performed, and the periodic system timer (systick) is installed.

The systick period is 1ms.

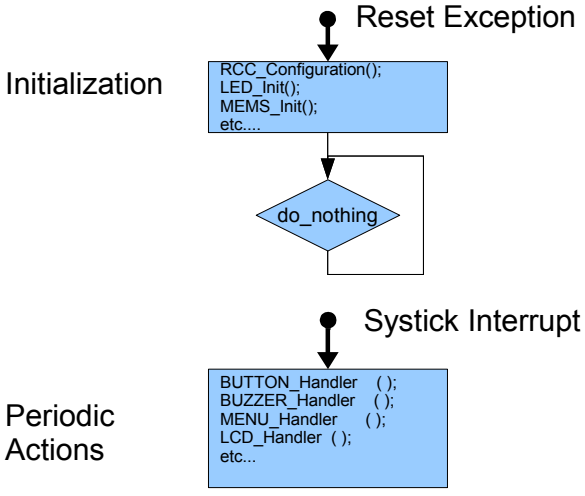
5.2.2 Periodic systick interrupt

The systick interrupt makes a call to the CircleOS systick interrupt handler which performs a short processing on each of the EvoPrimer components: LEDs, button, buzzer, LCD and so on.

5.2.3 Application scheduler

CircleOS is the base application of your EvoPrimer. It handles the MEMS input and menu selections and reacts to user actions.

When an application is run (usually through a menu selection), CircleOS calls an initialization routine for the application, then repeatedly calls the application handler at the Systick frequency until it returns a `MENU_LEAVE` value.



5.3 CircleOS resource usage

CircleOS offers lots of services and functions. When a CircleOS application is built, the linker removes unnecessary parts from the produced executable.

5.3.1 ROM usage

The CircleOS run-time, which performs device and peripheral initialization, task management, LCD, RTC, menu handling and various other functions requires around 22KB of code. This leaves around 10KB available for your own application.

If your application only uses features that are already in use by the CircleOS run-time, you just need ROM space for your application logic, no space will be required for the CircleOS services/APIs, as they are already linked in your application.

If you need some CircleOS services/APIs that the run-time does not itself use, then some of your ROM space (your “private” 10KB) is used to map these CircleOS services/APIs.

Note: You must be careful in the way you develop CircleOS applications for STM8L, so that your code does not consume too much ROM space. For instance, using “float” or “long” variables will require usage of expensive C runtime functions, which will consume a lot of space.

5.3.2 RAM usage

In the same way as ROM space, your application shares the RAM space with the CircleOS run-time. The CircleOS RAM usage is around 1KB, so you have 1KB available for your application.

5.3.3 EEPROM usage

CircleOS uses the STM8L EEPROM addresses 0x1000 to 0x1040 (64 bytes) to store information about the applications installed in its I2C (external) EEPROM.

Your application can freely use the EEPROM space above 0x1040 up to address 0x13FF.

Note: The I2C (external) 128KB EEPROM is used by CircleOS to store applications. It should not be used by your applications.

5.4 Managing applications on your EvoPrimer

Your EvoPrimer can store 4 applications in its 128KB I2C EEPROM. You can change the active application through the CircleOS interface by selecting the “Apps” entry from the main menu. This lets you select which application image should be loaded in the device Flash ROM.

The applications stored in the I2C EEPROM can be managed from your PC by the **Circle_STM8_mgr.exe** utility which can be found in the *[RIDE7_INSTALL_DIR]\Bin* directory.

This utility dedicated to the EvoPrimer allows you to:

- Display information about the embedded RLink programmer.
- List of the currently loaded CircleOS applications.
- Add new CircleOS applications.
- Remove CircleOS applications.
- Check how much I2C EEPROM space is available.

The following commands are available with the **Circle_STM8_mgr.exe** utility:

Command	Syntax	Description

List	L	List the loaded applications.
		`Circle_STM8_mgr.exe L`
		
		`Reading FAT table...`
		`App00: Name=Toggle`
		`... OK`
		
		`Number of free space for applications = 3`

Add	Afilename	Add a new application (HEX file).
		`Circle_STM8_mgr.exe Ac:\tmp\Toggle`
		
		`Programming application Toggle to EEPROM... OK`
		
		`Registering application in FAT... OK`
		
		`Reading FAT table...`
		`App00: Name=Toggle`
		`... OK`
		Note that you should *not* write the .HEX file extension

Erase	E*
	Eappname
		Remove an application (or all of them)
		`Circle_STM8_mgr.exe EToggle  /* Remove only 'Toggle' */`
		`Circle_STM8_mgr.exe E*  /* Remove ALL apps */`

Wait	W	When a command list is launched through a batch file, the W command allows you to pause the execution and to check the intermediate results.
Start	S	Start CPU execution

The options can be combined on a single command-line, and are executed sequentially. For instance, it is possible to remove all apps, then add the “Toggle” application, list the contents of EEPROM and start execution on the EvoPrimer with the following command:

```
Circle_STM8_mgr.exe E* Atoggle L S
```

5.5 Selecting the current application

One application is considered to be the “current application”. The ID of the current application is saved in the backup memory. From the main menu, you can launch it directly.

To change the current application, select the “Application” command from the main menu. Then select the application you wish to specify it as the “current application” and push the button. The new “current application” name will now appear in the main menu.

5.6 Downloading new applications

You will find at *www.stm8circle.com/projects/* a database where members can share their applications with the CircleOS community. An application can include both the source files and the HEX file, or the plain HEX file.

5.7 Restoring the factory configuration

If you have been experimenting with CircleOS applications and have modified your EvoPrimer configuration, you may want to restore the initial (factory) configuration.

In order to do this, please follow these steps:

1. Connect your EvoPrimer’s debug USB port to your PC.
2. Power-up your EvoPrimer by pressing its button.
3. Open a command prompt from Windows (Navigate to **Start | Programs | Accessories | Command prompt**)
4. Change the current directory to the Ride7 installation directory. This can be done with the following command (adapt it to your actual configuration):

```
cd "C:\Program Files\Raisonance\Ride"
```

5. Now change the current directory to the “Bin” directory. This can be done with the following command:

```
cd Bin
```

6. Erase your EvoPrimer, reprogram it with its factory ROM image and restart the device. These operations can be done with the following single command, which takes about 30 seconds:

```
Restore_Open4_STM8L_Circle_Factory.bat
```

5.8 Resetting your EvoPrimer

A hardware reset will restart your EvoPrimer (restoring it to a clean state), but will not remove its ROM contents.

- Remove any USB cable.
- Remove the plastic case protecting the STM8L152 target board.
- Remove the VCC-DB jumper, then replace it.
- Put the plastic case back in place.
- Restart your EvoPrimer by pressing its button.

6. Developing CircleOS applications

The full source files of CircleOS are available on the *http://www.stm8circle.com/* web site.
Once registered, you can download them, along with many resources for developing your application.

6.1 Developing your first CircleOS application

Creation of a CircleOS application is done automatically in Ride7:

1. Navigate to **Project | New project**.
2. Leave the **Type** selection list to **New application**.
3. In the **Processor** selection list, select the **STM8L15_CircleOS** device from the ST7-STM8 family, sub-family **Primer**.
4. Select an application name such as **My CircleOS application**.
5. Define the location where your new project will be created.
6. Click the **Finish** button
7. Your new project will be created, with an application containing a CircleOS application skeleton as well as the *CIRCLEOS_STM8L15.LIB* library and *stm8_ob.asm* option bytes file necessary to build your application with CircleOS.
8. Open the *application.c* file.
9. Search for the `Application_Name` variable in the file.
10. Change the `Application_Name` value from **`My App`** to **`HELLO`**.
11. In the `Application_Handler` function, create a new string as follows:
    ```
    const char msg[] = "Hello, World!";
    ```
12. Use the `DRAW_DisplayString` CircleOS service to display the `msg` string variable you just created:
    ```
    DRAW_DisplayString(5,20,msg,sizeof(msg)); // X, Y, string, length
    ```
13. Build your project using the **Project | Make Project** command.
14. Connect your EvoPrimer to your PC using the USB cable (be sure to use the “debug” USB port of your EvoPrimer).
15. From Ride7, go to **Debug | Start**, which will program your application to your EvoPrimer. This may take about 10 seconds.
16. Go to **Debug | Run**.
17. On your EvoPrimer, select your application name on the main menu.

Your application is now on your EvoPrimer.

For further information about CircleOS application programming and available OS services, please visit *http://www.stm8circle.com/.*

6.2 Libraries

Some common services are offered to ease your development of CircleOS applications.

1. The STM8L firmware libraries, written by ST, that provide access to the embedded peripherals (such as timers, ADC, communication interfaces, GPIO, etc...) of the STM8L microcontroller.
2. The low-level CircleOS functions that provide an easy access to the EvoPrimer's on-board peripherals: 3D accelerometer, LCD monitor, button, buzzer, battery, LEDs.
3. The graphical functions that provide powerful high-level functionality: Menu management, pointers (linked to the 3D accelerometer), character maps, sound.

The source files of these libraries can be found:

- On the CD or on the ST web site for the STM8L libraries. Specific documentation about the STM8L library is also available.
- On the *http://www.stm8circle.com/* web site for the CircleOS libraries (registration required).

Documentation about these libraries is accessible from Ride7.

6.3 Sharing your application with the Circle community

Once your application works properly, you can share it with the other members through the *http://www.stm8circle.com/* community.

7. Conformity and recycling

For more information on conformity and recycling, please visit the Raisonance website *www.raisonance.com*

ROHS Compliance (Restriction of Hazardous Substances)

Raisonance products are certified to comply with the European Union RoHS Directive (2002/95/EC) which restricts the use of six hazardous chemicals in its products for the protection of human health and the environment.

The restricted substances are as follows: lead, mercury, cadmium, hexavalent chromium, polybrominated biphenyls (PBB), and polybrominated diphenyl ethers (PBDE).

CE Compliance (Conformité Européenne)

Raisonance products are certified to comply with the European Union CE Directive.

In a domestic environment, the user is responsible for taking protective measures from possible radio interference the products may cause.

FCC Compliance (Federal Communications Commission)

Raisonance products are certified as Class A products in compliance with the American FCC requirements. In a domestic environment, the user is responsible for taking protective measures from possible radio interference the products may cause.

WEEE Compliance (The Waste Electrical & Electronic Equipment Directive)

As part of our continuing efforts to proved the best service to our users an communities, Raisonance is actively defining take-back and recycling programs for EU customers as part of the implementation of the European WEEE directive. The WEEE symbol is a reminder not to dispose of your electronic equipment in standard trash receptacles.

For more information about disposal and recycling of electronic goods, please refer to http://www.raisonance.com/support/weee.php.
Raisonance disposes of its electrical equipment according to the WEEE Directive (2002/96/EC).

Upon request, Raisonance can recycle customer's redundant products.

8. Glossary

Term	Description
STM8L152 Primer	EvoPrimer for STM8L152 MCU, a microcontroller evaluation & development platform
EvoPrimer	Commercial products which are derived from the Raisonance Open4 and distributed by STMicroelectronics for exploring, evaluating, developing and fine tuning applications for a variety of ST 8 and 32-bit microcontrollers. Certain use limitations may apply to these commercial products
Open4	Raisonance versatile hardware platform, which is used in a variety of commercial products that allow users to explore, evaluate, develop and fine tune applications for a range of microcontrollers that are mounted on Open4 compatible target boards. Also refered to as the "Base platform." Also distributed under the name EvoPrimer base
Target board	Boards featuring target microcontrollers which, when installed on an Open4-derived base allow execution of applications, programming of the target microcontroller and debugging of the applications running on the target microcontroller. Hardware features of target boards will vary depending on the target MCU and its features. Some features of the base platform may not be available depending on the features and capabilities of the target microcontroller
Extension board	Application-specific extension board, connected to the extension connector.
CircleOS	Embedded OS running on the Primer development kits.

9. Index

Alphabetical Index

ADC....8
application-specific extension boards....11
Autorun....15
Breakout application....14
BubLevel application....15
CE....23
Circle_STM8_mgr.exe....19
CircleOS architecture....17
CISC MIPS....8
communication peripherals....8
Compiling....15
Compliance....23
Conformity....23
current application....20
DAC....8
debugging....15
Directive....23
DMA....8
EEPROM....8 sv
EvoPrimer....4
Extension connector....10
factory configuration....20
FCC....23
Flash....8
hardware reset....20
impossible to recharge....8
Insert a target board....12
LCD....8
LCD display....9
Lead....23
LED (L0)....13
LED (L1)....13
Li-Ion....8
LIS3LV02DL....8 sv
MEMS accelerometer....8
MENU_LEAVE....17
Open4....4
oscillators....8
Power supply....9
programming....15
push button....9
Raisonance 23
RAM....8
reset....20
Ride7....15
RKit-STM8....15
Rkit-STM8 15
ROHS....23
RTC....8
Standalone operation....14
STLM20W87F....9
STM8L152C4T6....8
SWIM....8
systick....17
temperature sensor....9
TempLP application....14
Test....15
Time....15
timers....8
Toggle_with_CircleOS....16
Using your EvoPrimer applications....13
VCC_DB....8
watchdogs....8
Web site....4
WEEE....23
9

10. History

Date	Modifications
June 2010	Initial version.
July 2010	Notes added in Sections 3.3.2 and 4.2.2.

Disclaimer

Information in this document is subject to change without notice and does not represent a commitment on the part of the manufacturer. The software described in this document is provided under license and may only be used or copied in accordance with the terms of the agreement. It is illegal to copy the software onto any medium, except as specifically allowed in the license or non disclosure agreement.

No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or information storage and retrieval systems, for any purpose other than the purchaser's personal use, without prior written permission.

Every effort has been made to ensure the accuracy of this manual and to give appropriate credit to persons, companies and trademarks referenced herein.

This manual exists both in paper and electronic form (pdf).

Please check the printed version against the .pdf installed on the computer in the installation directory, for the most up-to-date version.

The examples of code used in this document are for illustration purposes only and accuracy is not guaranteed. Please check the code before use.

Copyright © Raisonance 1987-2010 All rights reserved